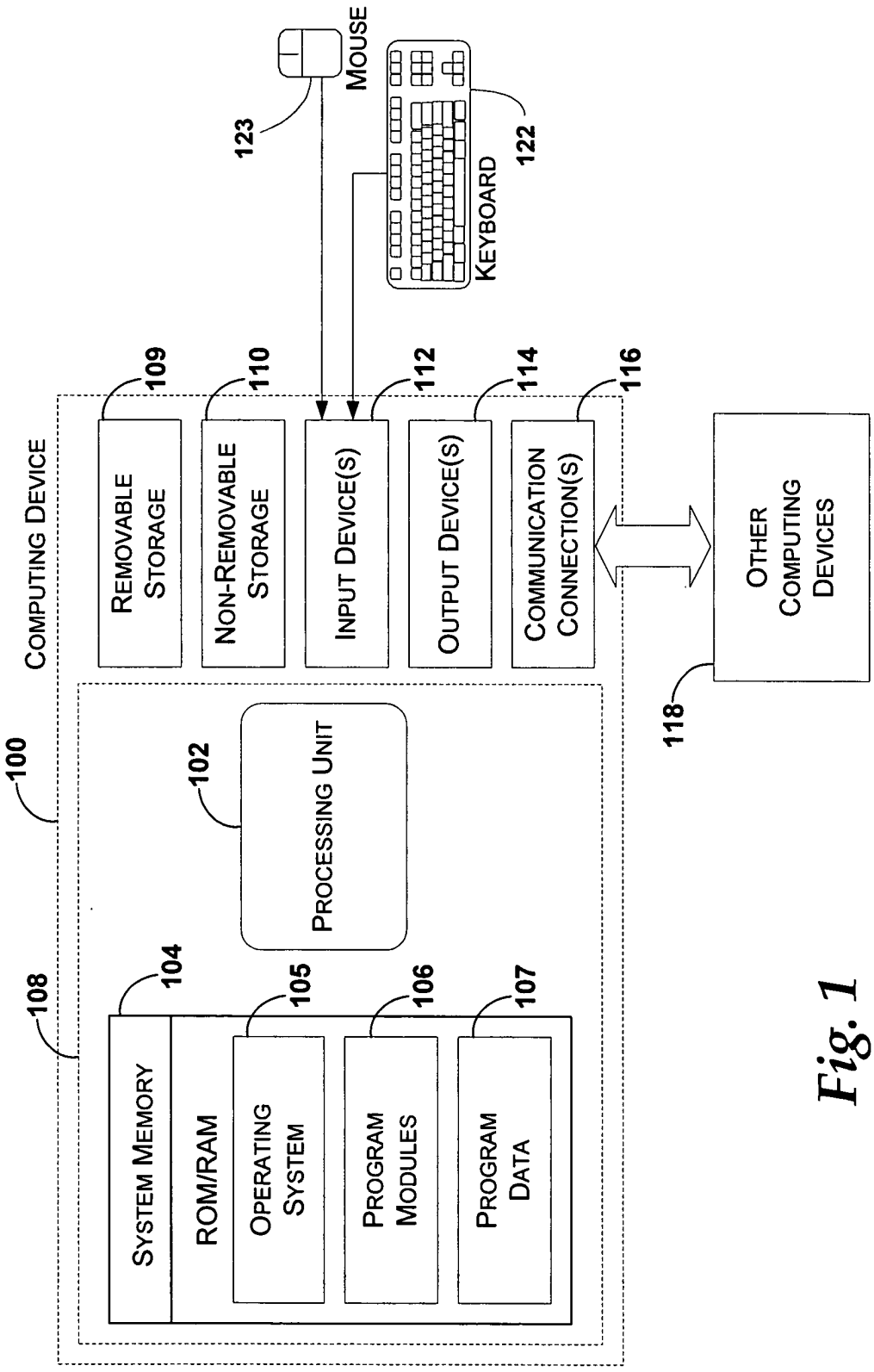
MOUSE
123
KEYBOARD
122
109
110
112
114
116
COMPUTING DEVICE
REMOVABLE STORAGE
NON-REMOVABLE STORAGE
INPUT DEVICE(S)
OUTPUT DEVICE(S)
COMMUNICATION CONNECTION(S)
OTHER COMPUTING DEVICES
118
100
102
PROCESSING UNIT
108
104
105
106
107
SYSTEM MEMORY
ROM/RAM
OPERATING SYSTEM
PROGRAM MODULES
PROGRAM DATA

Fig. 1

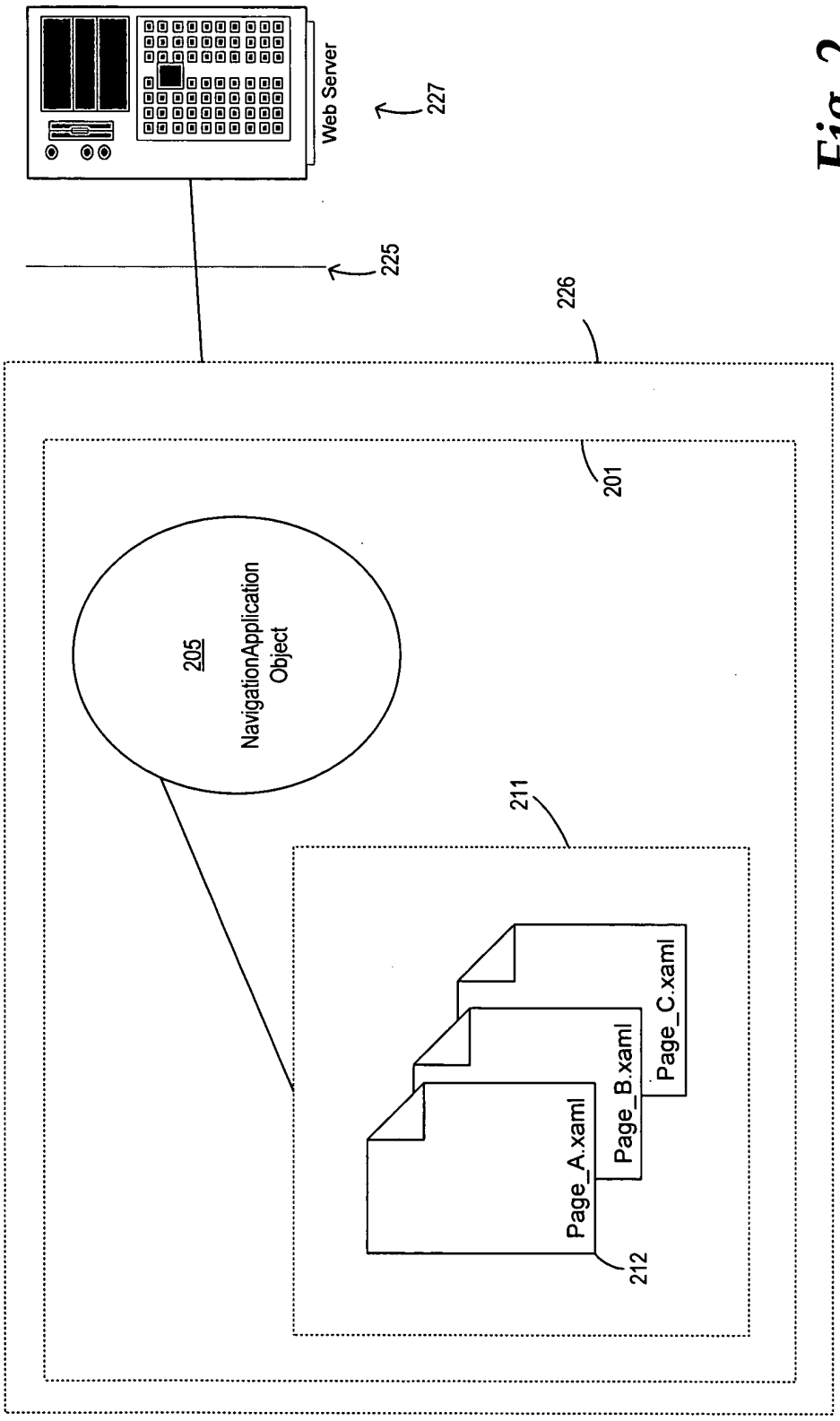
Web Server
227
225
226
201
205
NavigationApplication
Object
211
Page_A.xaml
Page_B.xaml
Page_C.xaml
212

Fig. 2

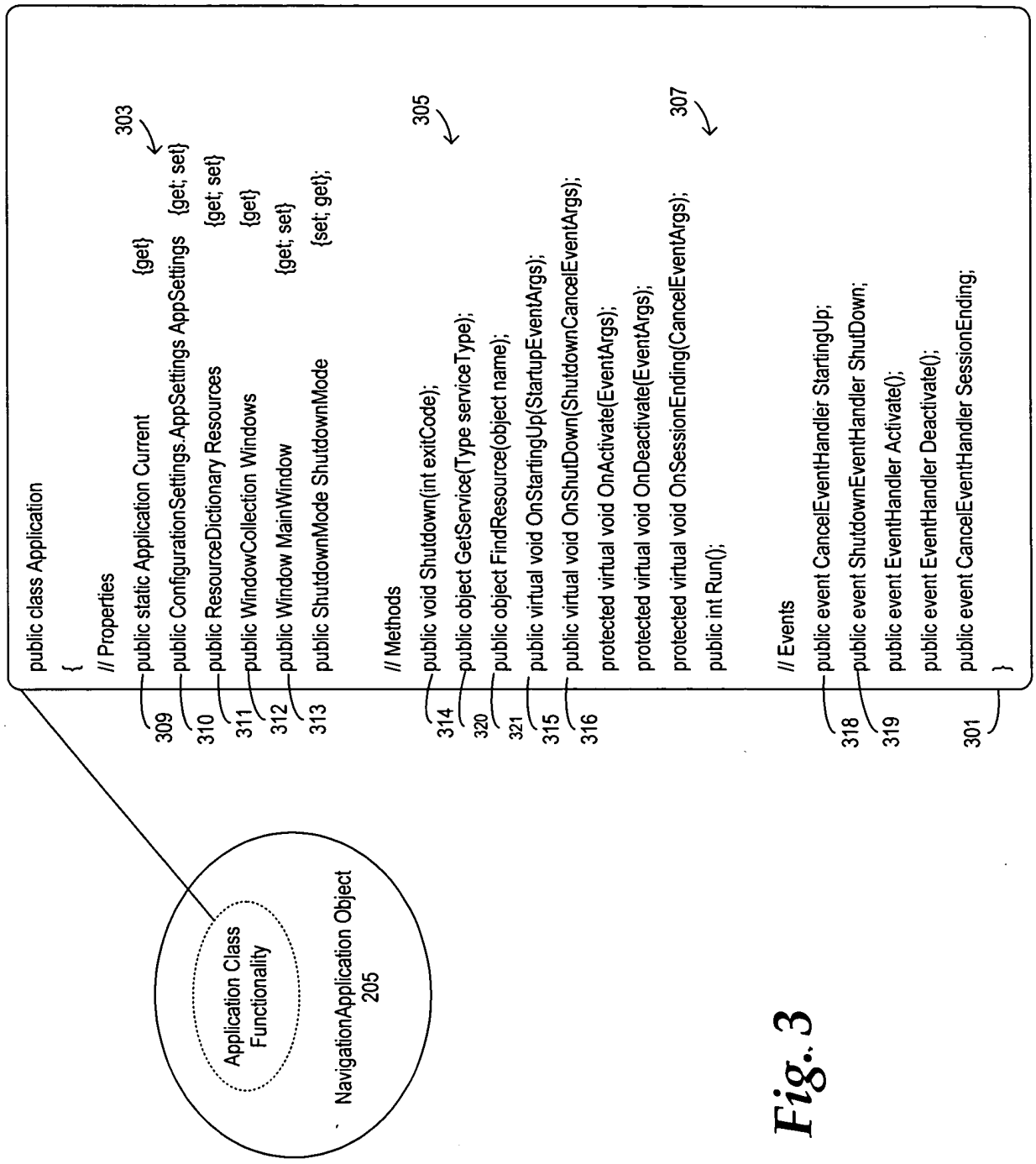
public class Application
{
// Properties
303
309 public static Application Current {get}
310 public ConfigurationSettings.AppSettings AppSettings {get; set}
311 public ResourceDictionary Resources {get; set}
312 public WindowCollection Windows {get}
313 public Window MainWindow {get; set}
public ShutdownMode ShutdownMode {set; get};
// Methods
305
314 public void Shutdown(int exitCode);
320 public object GetService(Type serviceType);
321 public object FindResource(object name);
315 public virtual void OnStartingUp(StartupEventArgs);
316 public virtual void OnShutDown(ShutdownCancelEventArgs);
protected virtual void OnActivate(EventArgs);
protected virtual void OnDeactivate(EventArgs);
307
protected virtual void OnSessionEnding(CancelEventArgs);
public int Run();
// Events
318 public event CancelEventHandler StartingUp;
319 public event ShutdownEventHandler ShutDown;
public event EventHandler Activate();
public event EventHandler Deactivate();
301 public event CancelEventHandler SessionEnding;
}
Application Class Functionality
NavigationApplication Object 205

Fig. 3

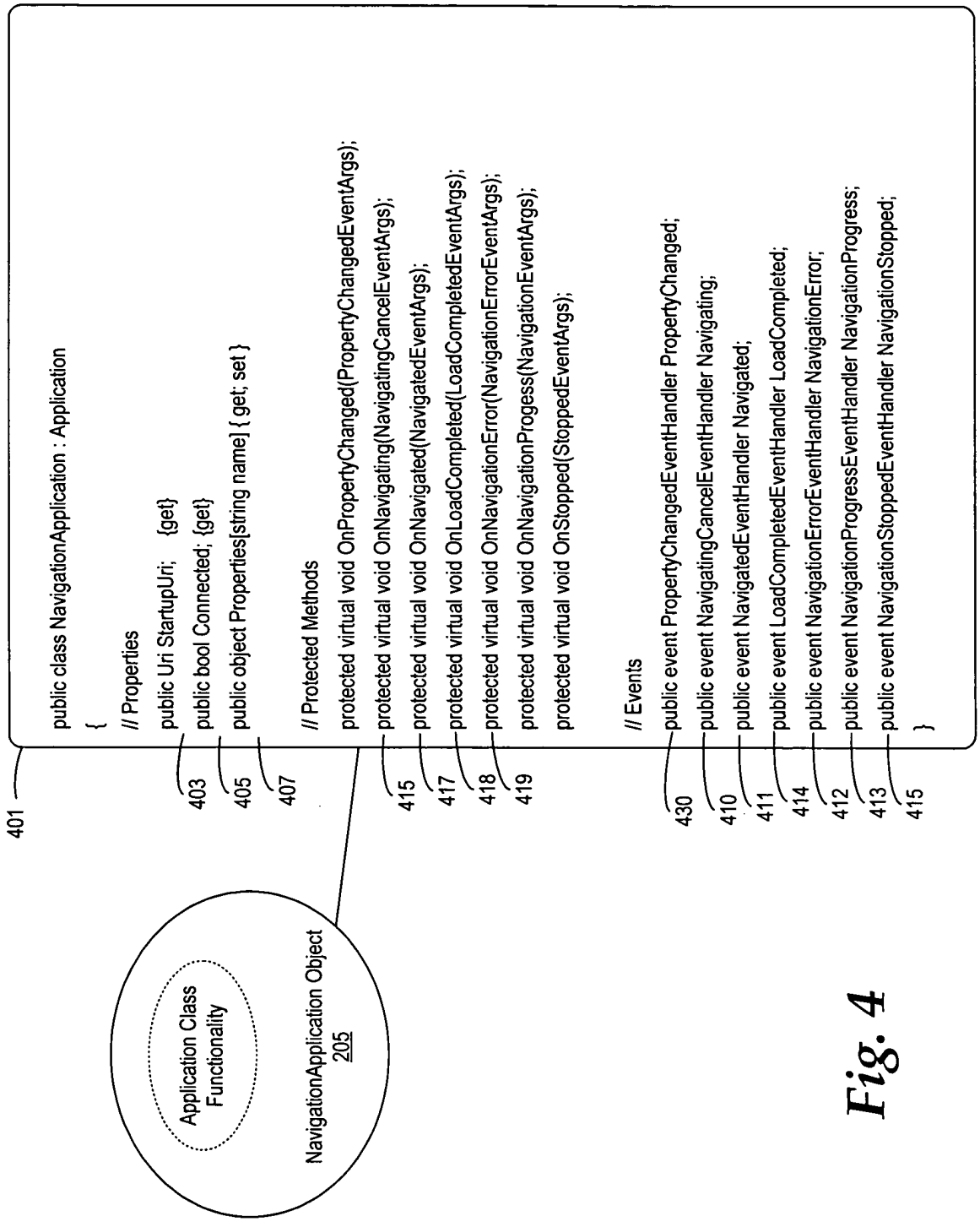
401
public class NavigationApplication : Application
{
// Properties
403
public Uri StartupUri; {get}
405
public bool Connected; {get}
407
public object Properties[string name] { get; set }
// Protected Methods
415
protected virtual void OnPropertyChanged(PropertyChangedEventArgs);
417
protected virtual void OnNavigating(NavigatingCancelEventArgs);
418
protected virtual void OnNavigated(NavigatedEventArgs);
419
protected virtual void OnLoadCompleted(LoadCompletedEventArgs);
protected virtual void OnNavigationError(NavigationErrorEventArgs);
protected virtual void OnNavigationProgess(NavigationEventArgs);
protected virtual void OnStopped(StoppedEventArgs);
// Events
430
public event PropertyChangedEventHandler PropertyChanged;
410
public event NavigatingCancelEventHandler Navigating;
411
public event NavigatedEventHandler Navigated;
414
public event LoadCompletedEventHandler LoadCompleted;
412
public event NavigationErrorEventHandler NavigationError;
413
public event NavigationProgressEventHandler NavigationProgress;
415
public event NavigationStoppedEventHandler NavigationStopped;
}
Application Class Functionality
NavigationApplication Object 205

Fig. 4

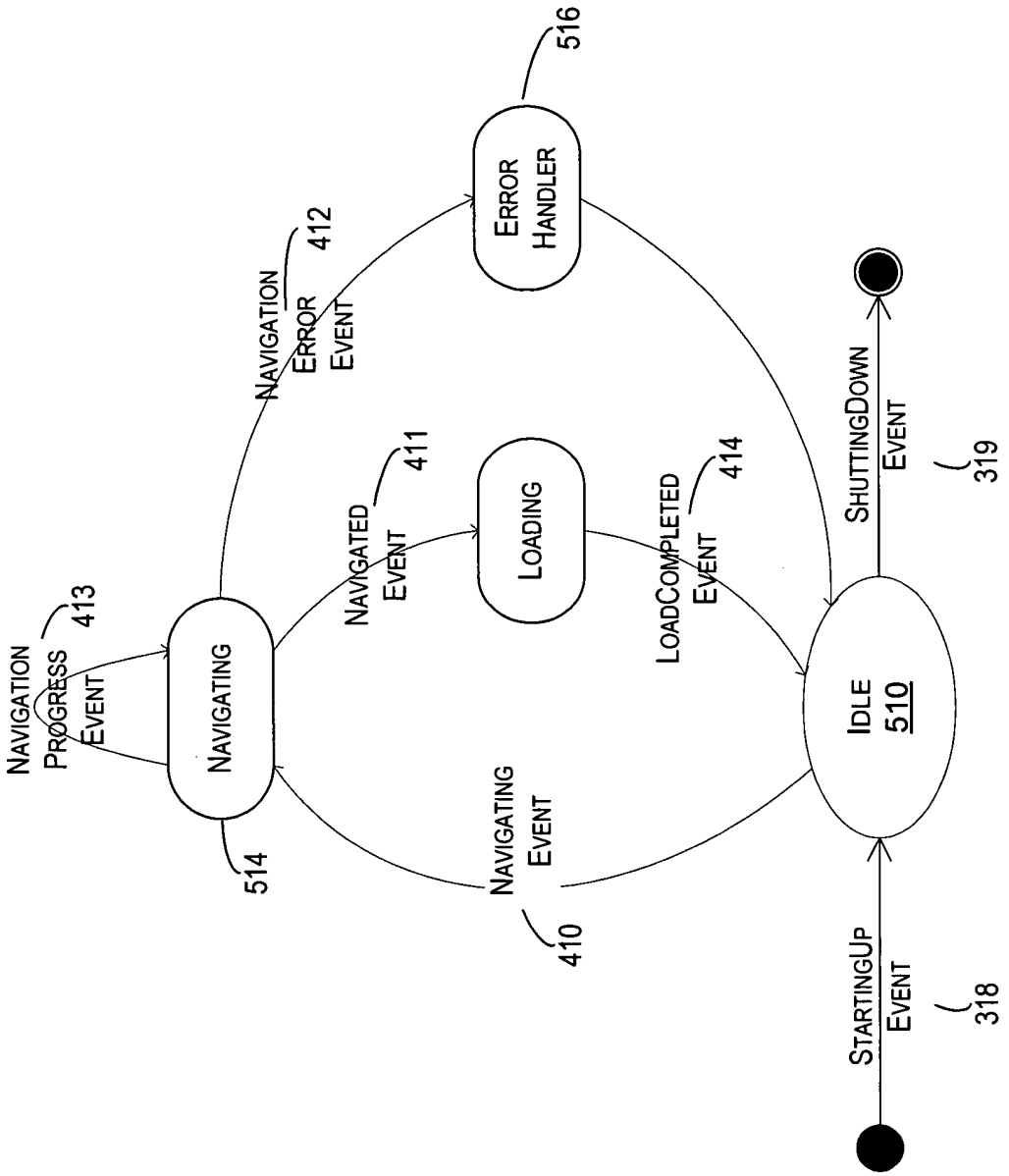
516
ERROR HANDLER
NAVIGATION ERROR EVENT
412
NAVIGATED EVENT
411
LOADING
LOADCOMPLETED EVENT
414
SHUTTINGDOWN EVENT
319
NAVIGATION PROGRESS EVENT
413
NAVIGATING
514
NAVIGATING EVENT
410
IDLE
510
STARTINGUP EVENT
318

FIG. 5

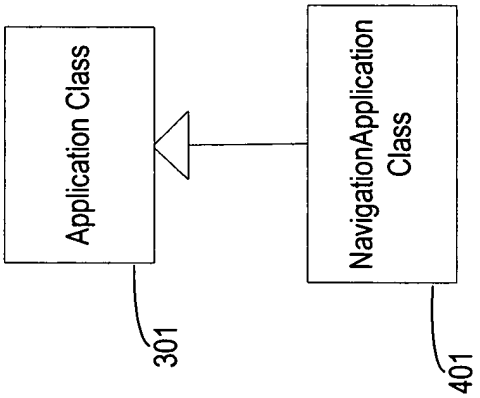

FIG. 6